DESCRIPTION

RECORDING SYSTEM, DATA RECORDING APPARATUS,
MEMORY APPARATUS, AND DATA RECORDING METHOD

Technical Field

5   The invention relates to a recording system,
a data recording apparatus, a memory apparatus, and
a data recording method, in which a memory card
which is detachable to/from an apparatus is used as
a recording medium.

10   Background Art

        According to an electrically rewritable
non-volatile memory called EEPROM (Electrically
Erasable Programmable ROM), since one bit is
constructed by two transistors, an occupation area

15   per bit is large and there is a limitation in case
of raising an integration degree. To solve such a
problem, a flash memory in which one bit can be
realized by one transistor by an all-bit batch
erasing method has been developed. The flash memory

20   is expected as a memory which can be substituted for
a recording medium such as magnetic disk, optical
disk, or the like.

        It is known that a memory card having a
flash memory is constructed so as to be detachable

25   to/from an apparatus. By using such a memory card,
a digital audio recording and reproducing apparatus
using the memory card in place of the conventional

1

disk-shaped medium such as CD (Compact Disc), MD (Minidisc), or the like can be realized. Besides audio data, still image data and motion image data can be also recorded into the memory card and it can

5   be used as a recording medium of a digital still camera or a digital video camera.

According to the flash memory, a data unit called a segment is divided into a predetermined number of clusters (fixed length) and one cluster is

10   divided into a predetermined number of sectors (fixed length). The cluster is also called a block. The sector is also called a page. In the flash memory, an erasure is performed in a lump on a cluster unit basis, and the writing or reading

15   operation is performed in a lump on a sector unit basis.

For example, in case of the flash memory of 4 MB (megabytes), as shown in Fig. 12, one segment is divided into 512 clusters. The segment is a unit

20   for managing a predetermined number of clusters. One cluster is divided into 16 sectors. One cluster has a capacity of 8 kB (kbytes). One sector has a capacity of 512 B. A memory of a capacity of 16 MB can be constructed by using four segments each

25   having a capacity of 4 MB.

As shown in Fig. 13A, logic cluster addresses are allocated to a memory space of 16 MB.

2

The logic cluster address is set to a length of 2
bytes in order to distinguish 512 × 4 = 2048
clusters. In Fig. 13A, the logic cluster address is
expressed by a hexadecimal number. 0x denotes the
5   hexadecimal notation. A logic address is an address
which is logically handled by a data processing
apparatus (software). A physical address is added
to each cluster in the flash memory. A
correspondence relation between the clusters and the
10   physical addresses is unchanged.

       According to the flash memory, by rewriting
data, an insulating film deteriorates and the number
of rewriting times is limited. Therefore, it is
necessary to prevent a situation such that accesses
15   are repetitively and concentratedly performed to a
certain same memory area (cluster). Therefore, in
case of rewriting data in a certain logic address
stored in a certain physical address, in a file
system of the flash memory, updated data is not
20   rewritten to an unused cluster. Thus, the
correspondence relation between the logic addresses
and the physical addresses before the data updating
changes after the updating. By performing such a
swapping process as mentioned above, the situation
25   such that the accesses are repetitively and
concentratedly performed to the same cluster is
prevented, so that a life of the flash memory can be

3

extended.

Since the logic cluster address is accompanied by the data which has once been written into the cluster, even if physical cluster addresses in which the data before updating and the data after the updating are written are changed, the same address is seen from a file management system and the subsequent accesses can be properly performed. Since the correspondence relation between the logic addresses and the physical addresses is changed by the swapping process, a logical/physical address conversion table showing the correspondence between them is needed. By referring to such a table, the physical cluster address corresponding to the designated logic cluster address is specified, thereby enabling the access to the cluster shown by the specified physical cluster address to be performed.

The logical/physical address conversion table is stored in a memory by the data processing apparatus. If a memory capacity of the data processing apparatus is small, the table can be stored in the flash memory. Fig. 13B shows an example of a logical/physical address conversion table regarding segment 1. As shown in Fig. 13B, in the logical/physical address conversion table, the physical cluster addresses (2 bytes) are made to

correspond to the logic cluster addresses (2 bytes) arranged in the ascending order, respectively. The logical/physical address conversion table is managed every segment and its size increases in accordance

5   with the capacity of the flash memory.

There is a case where it is desirable to set a data writing speed to be higher than the ordinary one by making a plurality of storages of the flash memory operative in parallel. For example,

10   an electronic music distribution EMD for distributing music data through a network is being put into practical use. The distributed music data is stored into a hard disk of a personal computer, data of a desired music piece is copied or moved

15   into a memory card by the personal computer, and the memory card is attached into a portable recorder, so that the user can easily listen to the desired music at a place other than his home. Data of a plurality of music pieces is downloaded into the memory card

20   from the hard disk by a parallel writing operation (at a high speed) and, upon reproduction, the music data is read out from the memory card at a normal speed.

Fig. 14 shows a construction of a

25   conventional logic address for four storages. In the example of the diagram, address spaces in the memory are expressed by 11 bits of A0, A1, …, and

A10. A0 denotes the LSB (least significant bit) and

A10 indicates the MSB (most significant bit). The

storages each having a capacity of 4 MB are switched

by the MSB (A10) and the second MSB (A9). Addresses

5   of 9 bits of A0 to A8 are allocated to a sector and

a segment in each storage.

When data is written, the operation is

executed at a timing as shown in Fig. 15. First,

the data is transferred from the host side to a page

10   buffer of a sector size. Time T is required to

transfer. In a next write busy period, the data is

transferred from the page buffer into a flash buffer

in the flash memory and the data is written into the

storage.

15   Upon reading, as shown in Fig. 16, the data

is read out from the flash memory for a read busy

period. The read-out data is transferred to a page

buffer of a sector size. In the next transfer time

T, the data is transferred from the page buffer to

20   the host side.

Fig. 17 is a flowchart showing a flow of

processes in case of writing data into continuous

logic sectors 0 to 3 belonging to different clusters

in a certain segment. In first step S11, a

25   logical/physical conversion table is formed with

respect to a segment as a target to be written. In

step S12, sector 0 is sent from the host side. The

6

time T is required for this transfer. In step S13, sector 0 is written into the flash memory. In step S14, sector 1 is sent from the host side. In step S15, sector 1 is written into the flash memory.

5   Processes for sending of sector 2 (step S16), writing of sector 2 (step S17), sending of sector 3 (step S18), and writing of sector 3 (step S19) are sequentially performed. Hitherto, for example, even if four storages are provided in parallel, since

10   accesses are concentrated to one storage, a high processing speed cannot be realized.

As for a data construction of one sector on the flash memory, as shown in Fig. 18, an area having a length of 16 bytes in which management

15   information is recorded is added to data of 512 bytes. The management information comprises a logic cluster number, cluster management information, and attribute information. The cluster management information is set to the same information among all

20   sectors in a certain cluster and includes information indicative of "valid/invalid" of the cluster or the like. The attribute information is information of every sector and includes copyright information or the like. For example, when the

25   flash memory is attached into the apparatus, the host side reads the management information and forms a table of the logic cluster and the physical

cluster with respect to the segment.

In case of making a plurality of storages operative in parallel, it is necessary to consider a method of accessing to a plurality of storages. Fig.

5 19 shows a construction of supplying addresses to four storages. Fig. 20 shows addresses in a flash memory of 4 MB × 4 = 16 MB. As described with reference to Fig. 14, the addresses are expressed by 11 bits of A0, A1, …, and A10. A0 is the LSB (least

10 significant bit) and A10 is the MSB (most significant bit). The storages each having a capacity of 4 MB are switched by the MSB (A10) and the second MSB (A9). The addresses of 9 bits of A0 to A8 are allocated to the clusters in each storage.

15 Hitherto, to switch the four storages, the addresses are supplied to the flash memory as shown in Fig. 19. The addresses A0 to A8 of 9 bits on the lower side are allocated in common to the four storages (0 to 3). The addresses A9 and A10 of 2

20 bits on the upper side are supplied to a 2-to-4 decoder 60. Selection signals CS0, CS1, CS2, and CS3 for selecting the storages are generated from the decoder 60.

In case of (A10, A9) = 00, the selection

25 signal CS0 to select the storage 0 is generated from the decoder 60. When (A10, A9) = 01, (A10, A9) = 10, or (A10, A9) = 11, each of the selection signals CS1,

8

CS2, and CS3 for selecting each of the storage 1, storage 2, and storage 3 is generated from the decoder 60.

An address change at the time when the address is increased from a state where all of 11 bits are equal to 0 to a state where they are equal to 1 due to the switching of the storages as mentioned above is shown by arrows in Fig. 20. That is, when the address changes from the head cluster of the storage 0 to the final cluster of the storage 0, the cluster address changes so as to subsequently shift to the head cluster of the storage 1. Fig. 21 shows an arrangement of the segments and the logic cluster addresses.

In the conventional switching of the storages of the flash memory mentioned above, a storage selection signal is formed by using a few bits from the MSB of the address. Thus, the segments are concentratedly arranged onto one storage and the segments are different every storage. According to such a method, a plurality of clusters of the same segment cannot be simultaneously written in parallel. For example, since four clusters (0x0004, 0x0005, 0x0006, 0x0007) in Fig. 21 are included in the same storage 0, they cannot be simultaneously written.

The clusters of a plurality of segments,

for example, four clusters (0x0004, 0x0204, 0x0404, 0x0604) in Fig. 21 can be simultaneously written into the storages 0 to 3. However, according to the flash memory, since a logical/physical address

5 conversion table is constructed every segment, it is necessary to refer to the logical/physical address conversion table upon accessing. Therefore, like an example mentioned above, when four clusters are simultaneously written over four segments, a memory

10 for holding address conversion tables of four segments is needed. Each time the data of one sector is written into each segment, the necessity of referring to the address conversion table is caused. Performance upon writing (or upon reading)

15 is deteriorated due to an overhead which is caused there.

It is, therefore, an object of the invention to provide a data processing system, a data processing apparatus, a memory apparatus, and a

20 data recording method, in which data can be written in parallel into a plurality of storages and performance upon reading can be improved.
Disclosure of Invention

To solve the above problems, according to

25 the invention of Claim 1, there is provided a data processing system in which a non-volatile memory apparatus having a plurality of storages in which

10

one cluster is constructed by a plurality of sectors
and one segment is constructed by a plurality of
clusters is detachable to/from a data processing
apparatus, wherein

5   the data processing apparatus has
           address designating means for designating
an address of the cluster in which data is recorded,
           the memory apparatus has
           recording means for recording the data into
10 the address designated by the address designating
means, and
           the data of a plurality of clusters in one
segment is distributed and arranged into a plurality
of storages.

15   According to the invention of Claim 2,
there is provided a data processing apparatus using
a non-volatile memory apparatus, as a recording
medium, having a plurality of storages in which one
cluster is constructed by a plurality of sectors and
20 one segment is constructed by a plurality of
clusters, wherein
           data of a plurality of clusters in one
segment is written into the memory apparatus so as
to be distributed and arranged into a plurality of
25 storages.

           According to the invention of Claim 3,
there is provided a non-volatile memory apparatus

11

which is detachable to/from a data processing

apparatus and has a plurality of storages in which

one cluster is constructed by a plurality of sectors

and one segment is constructed by a plurality of

5   clusters, wherein

        data of the plurality of clusters in one

segment is distributed and arranged into a plurality

of storages.

        According to the invention of Claim 6,

10   there is provided a data recording method of

recording, in parallel, data in which one cluster is

constructed by a plurality of sectors and which

exists over a plurality of clusters into a plurality

of storages, comprising the steps of:

15   designating a cluster address and writing

data into the designated cluster address; and

        after completion of the parallel writing

process, distributing and arranging the data of a

plurality of clusters in one segment into a

20   plurality of storages.

        According to the invention, the data of a

plurality of clusters in one segment can be written

in parallel. When the written data is read out, if

the data exists in the same segment, the switching

25   of the logical/physical address conversion table is

not caused, so that the performance upon reading can

be improved.

12

Brief Description of Drawings

  Fig. 1 is a block diagram showing a whole construction of an embodiment of the invention;

  Fig. 2 is a block diagram schematically showing a construction of a memory card in the embodiment of the invention;

  Fig. 3 is a block diagram showing a more detailed construction of the memory card in the embodiment of the invention;

  Fig. 4 is a schematic diagram for explaining an address construction in the embodiment of the invention;

  Fig. 5 is a schematic diagram for explaining the parallel writing operation in the embodiment of the invention;

  Fig. 6 is a timing chart for explaining the writing operation in the embodiment of the invention;

  Fig. 7 is a timing chart for explaining the reading operation in the embodiment of the invention;

  Fig. 8 is a flowchart for explaining the writing operation in the embodiment of the invention;

  Fig. 9 is a block diagram for explaining the switching of storages in the embodiment of the invention;

13

Fig. 10 is a schematic diagram for explaining the switching of storages in the embodiment of the invention;

Fig. 11 is a schematic diagram showing the relation between segments and logic cluster addresses in the embodiment of the invention;

Fig. 12 is a schematic diagram showing a construction of an example of a flash memory to which the invention can be applied;

Fig. 13 is a schematic diagram showing an example of a logical/physical address conversion table of the flash memory to which the invention can be applied;

Fig. 14 is a schematic diagram for explaining a conventional address construction;

Fig. 15 is a timing chart for explaining the conventional writing operation;

Fig. 16 is a timing chart for explaining the conventional reading operation;

Fig. 17 is a flowchart for explaining the conventional writing operation;

Fig. 18 is a schematic diagram for explaining a sector construction and management information;

Fig. 19 is a block diagram for explaining the conventional switching of storages;

Fig. 20 is a schematic diagram for

14

explaining the conventional switching of storages;
and

Fig. 21 is a schematic diagram showing the
conventional relation between segments and logic
5   cluster addresses.
Best Mode for Carrying Out the Invention

An embodiment of the invention will now be
described hereinbelow. Fig. 1 shows a construction
of a system to which the invention can be applied.
10   In the system, a data processing apparatus on the
host side and a memory card are connected via a
serial interface. In Fig. 1, reference numeral 1
denotes a CPU. A memory 2, a display 3, and an
input/output unit 4 are connected to a bus of the
15   CPU 1.

A serial interface 5 is arranged between
the CPU bus and a memory card 6 surrounded by a
broken line. The memory 2 includes an ROM for
storing a program and an RAM which is used as a work
20   area. Specifically speaking, the data processing
apparatus is a personal computer, a digital still
camera, a digital video camera, a digital audio
recorder, or the like. The memory card 6 has a
flash memory 7. The flash memory 7 is, for example,
25   an NAND-type flash memory (non-volatile memory).
There is also a case where an enciphering circuit is
assembled in the memory card 6 for the purpose of

15

protection of the copyright of contents which are stored.

The invention can be also applied to a case where the transmission and reception of data between the data processing apparatus and the memory card 6 are executed by a parallel interface instead of the serial interface.

The flash memory 7 is a memory having a capacity of, for example, 4 MB × 4 = 16 MB. As mentioned with reference to Fig. 12, in case of the flash memory of 4 MB (megabytes), one segment is divided into 512 clusters and one cluster is divided into 16 sectors. One cluster has a capacity of 8 kB (kbytes) and one sector has a capacity of 512 B. As described with reference to Fig. 13A, the logic cluster addresses are allocated into the memory space of 16 MB. As described with reference to Fig. 13B, the logical/physical address conversion table showing the correspondence relation between the logic cluster addresses and the physical cluster addresses is formed on a segment unit basis. Further, as mentioned above, the physical cluster addresses of 11 bits of A0, A1, …, and A10 are used for the flash memory of 4 MB × 4 = 16 MB.

As shown in Fig. 2, the parallel writing can be performed to the flash memory 7. Fig. 2 shows only the portion regarding the data

16

input/output for simplicity of explanation. Four
corresponding memory cells MC0 to MC3 are provided
for storages 0 to 3, respectively. Data is supplied
to the memory cells MC0 to MC3 through the data bus

5   and flash buffers BF0 to BF3, respectively. That is,
when the write data of one page is accumulated into
each of the flash buffers BF0 to BF3 through the
data bus, the data is simultaneously transferred
from the flash buffers BF0 to BF3 to the memory

10 cells MC0 to MC3. Although Fig. 2 shows an example
in which one IC package has four storages, four
flash memories of different packages can be also
used. Further, a plurality of flash memories each
having a plurality of storages in a package can be

15 also combined.

     Fig. 3 shows a more specific construction
of the memory card 6 to which the invention can be
applied. The memory card 6 is formed by
constructing a control block 11 and the flash memory

20 7 as one chip IC. The bidirectional serial
interface 5 between the CPU 1 of the data processing
apparatus and the memory card 6 comprises ten lines.
Four main lines are a clock line SCK for
transmitting a clock upon data transmission, a

25 status line SBS for transmitting a status, a data
line DIO for transmitting data, and an interruption
line INT. Two GND lines and two VCC lines are

provided as other power supplying lines. Two lines
Reserv are undefined lines.

The clock line SCK is the line for
transmitting the clock synchronized with the data.

5   The status line SBS is the line for transmitting a
signal indicative of the status of the memory card 6.
The data line DIO is the line for inputting and
outputting a command and enciphered audio data. The
interruption line INT is the line for transmitting

10   an interruption signal for requesting an
interruption to the CPU 1 of the data processing
apparatus from the memory card 6. The interruption
signal is generated when the memory card 6 is
attached. In the embodiment, however, since the

15   interruption signal is transmitted through the data
line DIO, the interruption line INT is connected to
the ground and is not used.

A serial/parallel conversion,
parallel/serial conversion, and interface cluster

20   (abbreviated to an S/P, P/S, I/F cluster) 12 in the
control block 11 is connected to the interface 5.
The S/P, P/S, I/F block 12 converts the serial data
received from the data processing apparatus into
parallel data, fetches it into the control block 11,

25   converts the parallel data from the control block 11
into the serial data, and sends it to the data
processing apparatus.

In a format by which data is transmitted
through the data line DIO, a command is first
transmitted and, thereafter, data is transmitted.
The S/P, P/S, I/F block 12 stores the command into a
5   command register 13 and stores the data into a page
buffer 14 and a write register 15.   An error
correction encoding circuit 16 is provided in
association with the write register 15.   The error
correction encoding circuit 16 forms a redundancy
10   code of an error correction code for the data
temporarily stored in the page buffer 14.
        The output data of the command register 13,
page buffer 14, write register 15, and error
correction encoding circuit 15 is supplied to a
15   flash memory interface and sequencer (abbreviated to
a memory I/F, sequencer) 17.   The memory I/F,
sequencer 17 is an interface between the control
block 11 and flash memory 7 and controls the
transmission and reception of the data between them.
20   The data is written into the flash memory 7 through
the memory I/F, sequencer 17.
        The data read out from the flash memory 7
is supplied to the page buffer 14, a read register
18, and an error correcting circuit 19 through the
25   memory I/F, sequencer 17.   The data stored in the
page buffer 14 is error corrected by the error
correcting circuit 19.   The error corrected outputs

19

of the page buffer 14 and read register 18 are

supplied to the S/P, P/S, I/F block 12 and supplied

to the CPU 1 of the data processing apparatus

through the serial interface 5.

5   Reference numeral 20 denotes a

configuration ROM in which version information,

various attribute information, and the like of the

memory card 6 have been stored. A switch 21 for

prevention of erroneous erasure which can be

10   operated by the user as necessary is provided for

the memory card 6. When the switch 21 is in a

connecting state of erasure inhibition, even if a

command to instruct the erasure of the flash memory

7 is sent from the data processing apparatus side,

15   the erasure of the flash memory 7 is inhibited.

Further, reference numeral 22 denotes an oscillator

for generating a clock serving as a reference for

timing of the process of the memory card 6.

The serial interface between the data

20   processing apparatus and memory card 6 in the

embodiment of the invention will be described

further in detail. When the data is read out from

the memory card 6, a read command is transmitted to

the memory card 6 from the data processing apparatus

25   and the memory card 6 receives the read command.

After completion of the transmission of the command,

the memory card 6 executes a process for reading out

the data in the address designated by the received
read command from the flash memory 7. While this
process is being executed, a busy signal (high
level) is transmitted to the data processing
5   apparatus through the data line DIO. After
completion of the reading of the data from the flash
memory 7, the output of the busy signal is stopped
and an output of a ready signal (low level) showing
the completion of preparation for sending the data
10   from the memory card 6 to the data processing
apparatus is started.

By receiving the ready signal from the
memory card 6, the data processing apparatus knows a
fact that the process corresponding to the read
15   command has been ready. The memory card 6 outputs
the data read out to the page buffer to the data
processing apparatus through the data line DIO. The
status in which each of the above processes is
executed is shown by a level change of the status
20   line SBS.

When the data is written into the flash
memory 7 of the memory card 6, a write command is
transmitted from the data processing apparatus to
the memory card 6 via the data line DIO. A write
25   address is transmitted in association with the write
command. Although the writing and reading
operations of the data are performed on a sector

21

unit basis in the flash memory 7, the file is managed on a cluster unit basis in the data processing apparatus and addresses from the data processing apparatus are based on a cluster unit.

5   Subsequently, the data processing apparatus transmits the write data to the memory card 6 through the data line DIO. In the memory card 6, the received write data is accumulated into the page buffer. When the transmission of the write data is

10   finished, the memory card 6 executes a process for writing the write data into the flash memory 7. A busy signal is outputted during the writing process. When the writing process of the write data is finished in the memory card 6, the output of the

15   busy signal is stopped and the ready signal (low level) is transmitted to the data processing apparatus.

In case of performing the parallel writing operation via the serial interface, a command, an

20   address, and data for writing into the storage 0 are transmitted and, thereafter, in a state where the busy signal is at the high level, a command, an address, and data for writing into the storage 1, a command, an address, and data for writing into the

25   storage 2, and a command, an address, and data for writing into the storage 3 are sequentially transmitted. The command, address, and data for

22

writing into the storage 0 are again transmitted.
At this time point, the previous writing process of
the data into the storage 0 has been finished and
the busy signal is at the low level. The parallel
5   writing can be performed by repeating such an
operation. The commands, addresses, and data can be
also simultaneously transmitted by a method of using
four serial interfaces in parallel.

The above embodiment of the invention will
10   be described further in detail. Fig. 4 shows a
construction of the address in the embodiment.
Address spaces in the memory are shown by 11 bits of
A0, A1, ..., and A10. A0 indicates the LSB (least
significant bit) and A10 indicates the MSB (most
15   significant bit). The storages each having a
capacity of 4 MB are switched by the LSB (A00) and
the second LSB (A1). The addresses of 9 bits of A2
to A10 are allocated to the sectors and segments in
the respective storages.

20   Fig. 5 is a diagram for explaining a file
managing method in the system of Fig. 1 using the
memory card 6 as a storing medium. In Fig. 5,
reference numeral 30 denotes data in a data file,
for example, in a compressed audio data file. As
25   for the compressed audio data, usually, a file is
formed every music piece and the file is recorded
into the flash memory 7 in the memory card 6 on a

23

sector unit basis and read out from the flash memory
7.

In case of recording such data 30 into the
flash memory 7 in parallel, as shown in Fig. 5,
5   writing sectors are selected from a plurality of
clusters so that the sectors are continuously
arranged in each cluster after the writing process
and the data is simultaneously written into the
selected sectors. Now, assuming that the size of
10   data 30 coincides with four clusters, the data 30 is
recorded into the four clusters in the flash memory
7.

As shown in Fig. 5, the data is recorded so
that the sectors are arranged in each cluster in
15   each storage in the original order after the writing.
For example, in the case where the sectors numbered
as 0, 1, 2, 3, … in the original order are written
into the storages 0 to 3 in parallel, the data of No.
0 is recorded into the head sector in the cluster in
20   the storage 0, the data of No. 16 is recorded into
the head sector in the cluster in the storage 1, the
data of No. 32 is recorded into the head sector in
the cluster in the storage 2, and the data of No. 48
is recorded into the head sector in the cluster in
25   the storage 3, respectively.

As mentioned above, when the data is
numbered every sector, four data units of the

24

numbers having offsets of the number which is equal
to the number of sectors of the cluster are
converted into parallel data and simultaneously
written into four storages. Thus, in each storage
5   in the flash memory 7, the data is arranged in the
original order into the cluster constructed in the
same storage in a manner similar to that of the
existing flash memory. Therefore, the compatibility
with the file format of the existing flash memory is
10 held.

The data is sequentially read out every
cluster from the flash memory recorded as mentioned
above. For example, the data is sequentially read
out from the head sector of the cluster in the
15 storage 0 in Fig. 5. Subsequently, the data is
sequentially read out from the head sector of the
cluster in the storage 1. In this manner, the data
is sequentially read out from the cluster in the
storage 2 and the cluster in the storage 3. The
20 order of the read-out data is the same as the
original order. The erasing operation is performed
on a cluster unit basis constructed every storage.

As mentioned above, according to the data
arrangement after the parallel writing in the
25 embodiment, the cluster is constructed in the same
storage in a manner similar to the existing flash
memory. Therefore, the compatibility of the file

25

format with that of the existing flash memory can be
held.

Fig. 6 shows the writing operation in the
embodiment. Data is first transferred from the host
5   side to the page buffer of the sector size. Further,
the data is transferred from the page buffer into
the flash buffer BF0 of the storage 0. The time T
is required to transfer. For the next write busy
period, the data is written from the flash buffer
10   BF0 into the storage 0. After the first transfer
period T, the data of the next sector is transferred
and written into the storage 1 for the write busy
period. Since the writing operation into the
storages 0 to 3 is performed in parallel as
15   mentioned above, the writing speed higher than that
in the conventional writing operation shown in Fig.
15 is accomplished.

Upon reading, as shown in Fig. 7, for a
read busy period, the data is read out from each of
20   the storages 0 to 3 and the read-out data is
transferred to the flash buffers BF0 to BF3 of the
sector size. At the next transfer time T, the data
is transferred from the flash buffer BF0 to the page
buffer and, further, the data is transferred from
25   the page buffer to the host side. Subsequently, the
data is sequentially outputted from the flash
buffers BF1, BF2, and BF3 to the page buffers and

26

the data is transferred from the page buffers to the host side. The reading speed higher than that in the process of the conventional reading operation in which the read busy operations are sequentially

5   performed (Fig. 16) is accomplished.

Fig. 8 is a flowchart showing a flow for processes in case of writing data into continuous logic sectors 0 to 3 belonging to the different clusters in a certain segment. In first step S1, a

10 logical/physical conversion table is formed with respect to a segment as a target to be written. In step S2, sector 0 is sent from the host side to the page buffer and the data in sector 0 is transferred from the page buffer to the flash buffer. The time

15 T is required to transfer. In next step S3, in parallel with the sending of sector 1, sector 0 is written into one storage in the flash memory in step S4.

In step S5, sector 2 is sent. In step S6,

20 sector 1 is written into one storage in the flash memory in parallel. Subsequently, processes for sending of sector 3 (step S7), writing of sector 2 (step S8), and writing of sector 3 (step S9) are similarly performed. According to the embodiment of

25 the invention, the situation such that the acceses are concentrated to one storage as in the conventional apparatus does not occur, and the

27

segments are not switched. Therefore, since there
is no need to form the logical/physical conversion
table, the high processing speed can be realized.

As an example of a storage switching method
5   at the time of performing the parallel writing
operation as mentioned above and reading out the
written data will be described. Fig. 9 shows a
construction for supplying addresses to four
storages in the embodiment. Fig. 10 shows physical
10 addresses in the flash memory of 4 MB × 4 = 16 MB.

As described with reference to Fig. 4, the
physical addresses are expressed by 11 bits of A0,
A1, …, and A10. A0 indicates the LSB (least
significant bit) and A10 shows the MSB (most
15 significant bit).

In the embodiment of the invention, when
the formed addresses A0 to A10 of 11 bits are sent
to the flash memory, the addresses A2 to A10 are
supplied to the flash memory as addresses for
20 specifying the addresses of the sectors and segments.
Two lower bits of A0 and A1 are supplied to the
flash memory as addresses for specifying the storage
switching. That is, as shown in Fig. 9, the
addresses A2 to A10 of nine bits on the upper side
25 are sent in common to the four storages (0 to 3).
The addresses A0 and A1 of two bits on the lower
side are supplied to a 2-to-4 decoder 40. Selection

28

signals CS0, CS1, CS2, and CS3 for selecting each

storage are generated from the decoder 40. When (A1,

A0) = 00, the selection signal CS0 to select the

storage 0 is generated from the decoder 40. When

5   (A1, A0) = 01, (A1, A0) = 10, or (A1, A0) = 11, the

selection signal CS1, CS2, or CS3 for selecting the

storage 1, storage 2, or storage 3 is generated from

the decoder 40, respectively. In the constructional

example of Fig. 3, the decoder 40 is provided in the

10   memory I/F, sequencer 17.

In case of performing the storage switching

as mentioned above, an address change at the time

when the physical address is increased from the

address in which all 11 bits are equal to 0 to the

15   address in which all 11 bits are equal to 1 is shown

by arrows in Fig. 10. That is, the address change

starts from the head cluster of the storage 0.

Subsequently, the head cluster of the storage 1 is

designated. When the address changes to the head

20   cluster of the storage 3 via the head cluster of the

storage 2, the physical cluster address changes so

that the cluster is shifted to the second cluster of

the storage 0.

Fig. 11 shows an arrangement of the

25   segments and the logic cluster addresses in the

embodiment of the invention. As will be understood

from the arrangement of the logic cluster addresses

29

in Fig. 11, 512 clusters included in one segment are
constructed by 128 clusters included in each of the
four storages. The logical/physical address
conversion table is formed every segment. Therefore,
5   if no segment is changed, the logical/physical
address conversion table to be referred to or
updated is not changed, so that a deterioration of
the reading performance due to the access to the
table or the updating of the table can be prevented.
10   The data can be simultaneously written into the
continuous logic cluster addresses, for example,
0x0004 to 0x0007.

When the logic cluster addresses are
discontinuous like 0x0000, 0x0200, 0x0400, and
15   0x0600, since those addresses exist in one storage,
the data cannot be simultaneously written. However,
since a probability that the process for writing the
continuous logic sectors into such discontinuous
cluster addresses actually occurs is very low, a
20   large problem will not occur.

Although the embodiment has been described
above with respect to the flash memory of 4 MB in
which one sector consists of 512 B, one cluster
consists of 8 kbytes, and one storage consists of
25   512 clusters, those values are shown as an example
and the invention can be also applied to flash
memories of other numerical values. For example,

30

the capacity of one cluster can be set to 16 kB.
The invention can be also applied to a flash memory
such that a capacity of one storage is equal to 8 MB
(1024 clusters × 8 kB), 16 MB (1024 clusters × 16

5   kB), 32 MB (2048 clusters × 16 kB), 64 MB (4096
clusters × 16 kB), or the like.

According to the invention, when data
existing over a plurality of clusters is written,
the data can be simultaneously written, so that the

10   high speed writing can be performed. When the data
existing over a plurality of clusters is written or
read out, if it exists in the same segment, the
switching of the logical/physical address conversion
table can be made unnecessary, so that a high

15   accessing speed can be realized.